AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions, and listings, of claims in the application.

1. (Currently Amended) A method of performing a two-dimensional discrete cosine transform (DCT) using a microprocessor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions, wherein the method comprises:

receiving a two-dimensional block of integer data having C columns and R rows, wherein each of the R rows contains a set of C row data values, wherein the block of integer data is indicative of a portion of an image, wherein each of C and R is an even integer; and

for each row,

loading the entire set of C row data values of the row into a set of C/2 registers of the microprocessor;

converting the C row data values into floating point form, wherein each of the registers holds [[each hold]] two of the floating point row data values; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations are performed using SIMD floating point instructions;

altering the arrangement of values in the registers;

performing a second plurality of weighted-rotation operations on the values in the registers;

again altering the arrangement of the values in the registers;

performing a third plurality of weighted-rotation operations on the values in the registers;

yet again altering the arrangement of the values in the registers; [[and]]

performing a fourth plurality of weighted-rotation operations on the values in the registers to obtain C intermediate floating point values; and

storing the C intermediate floating point values into a next available row of an intermediate buffer.

2. (Previously Presented) The method of claim 1, wherein said converting is accomplished using a packed integer word to floating-point conversion (pi2fw) instruction.

3. (Previously Presented) The method of claim 1, wherein said weighted-rotation operations are accomplished using a packed swap doubleword (pswapd) instruction, a packed floating-point multiplication (pfmul) instruction and a packed floating-point negative accumulate (pfpnacc) instruction.

4. (Cancelled)

5. (Cancelled) ~~The method of claim 1, further comprising:~~

~~for each row,~~

~~storing the intermediate floating point values to an intermediate buffer~~.

6. (Currently Amended) The method of claim 5, further comprising:

for two columns of the intermediate buffer at a time:

loading data from the two columns [[of intermediate data]] into [[each of]] a plurality of registers of the microprocessor so that each of the registers holds one value from a first of the two columns and one value from a second of the two columns, wherein the one value from the first of the two columns and the one value from the second of the two columns are taken from the same row of the intermediate buffer; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations for two columns are performed in parallel using SIMD floating point instructions.

7. (Previously Presented) The method of claim 6, wherein said weighted-rotation operations for two columns at a time are accomplished using a packed floating-point multiplication

(pfmul) instruction, a packed floating-point subtraction (pfsub) instruction and a packed floating-point addition (pfadd) instruction.

8. (Original) The method of claim 6, further comprising:

for two columns at a time,

as each weighted-rotation operation is done, storing weighted-rotation operation results to the intermediate buffer.

9. (Original) The method of claim 8, further comprising:

for two columns at a time,

retrieving weighted-rotation operation results from the intermediate buffer;

performing a second plurality of weighted-rotation operations on the retrieved values;

again storing weighted-rotation operation results to the intermediate buffer as the weighted-rotation operations of the second plurality are done;

again retrieving weighted-rotation operation results from the intermediate buffer;

performing a third plurality of weighted-rotation operations on the retrieved values;

yet again storing weighted-rotation operation results to the intermediate buffer as the weighted-rotation operations of the third plurality are done;

yet again retrieving weighted-rotation operation results from the intermediate buffer;

performing a fourth plurality of weighted-rotation operations on the retrieved values;

converting the weighted-rotation operation results from the fourth plurality to integer results.

10. (Original) The method of claim 9, further comprising:

for two columns at a time, writing the integer results to an output buffer.

11. (Currently Amended) A method of performing a discrete cosine transform (DCT) using a microprocessor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions, wherein the method comprises:

receiving a two-dimensional block of integer data having C columns and R rows, wherein each of C and R is an even integer, wherein the two-dimensional block represents a portion of an image; and

for two columns at a time,

loading column data from the two columns into registers of the microprocessor so that each of the registers holds one value from a first of the two columns and one value from a second of the two columns, wherein the one value from the first of the two columns and the one value from the second of the two columns are taken from the same row of the two-dimensional block;

converting the column data into floating point form [[, wherein the registers each hold a floating point column data value from two columns]]; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations for the two columns are performed in parallel using SIMD floating point instructions;

as each weighted-rotation operation is done, storing weighted-rotation operation results to an intermediate buffer.

12. (Previously Presented) The method of claim 11, wherein said weighted-rotation operations are accomplished using a packed floating-point multiplication (pfmul) instruction, a packed floating-point subtraction (pfsub) instruction and a packed floating-point addition (pfadd) instruction.

13. (Cancelled)

14. (Previously Presented) The method of claim 11, further comprising:

for two columns at a time,

retrieving weighted-rotation operation results from the intermediate buffer;

performing a second plurality of weighted-rotation operations on the retrieved values;

again storing weighted-rotation operation results to the intermediate buffer as the weighted-rotation operations of the second plurality are done;

again retrieving weighted-rotation operation results from the intermediate buffer;

performing a third plurality of weighted-rotation operations on the retrieved values;

yet again storing weighted-rotation operation results to the intermediate buffer as the weighted-rotation operations of the third plurality are done;

yet again retrieving weighted-rotation operation results from the intermediate buffer;

performing a fourth plurality of weighted-rotation operations on the retrieved values;

converting the weighted-rotation operation results from the fourth plurality to integer results.

15. (Original) The method of claim 14, further comprising:

for two columns at a time, writing the integer results to an output buffer.

16. (Currently Amended) A computer system comprising:

a processor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions; and

a memory coupled to the processor, wherein the memory stores software instructions executable by the processor to implement a two-dimensional discrete cosine transform method, the method comprising: receiving a <u>two-dimensional</u> block of integer data having C columns and R rows, wherein each of the R rows contains <u>a set of</u> C row data values, wherein the block of integer data is indicative of a portion of an image<u>, wherein each of C and R is an even integer</u>; and

for each row,

loading the entire set of C row data values of the row into a set of C/2 registers of the processor;

converting the C row data values into floating point form, wherein each of the registers holds [[each hold]] two of the floating point row data values; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations are performed using SIMD floating point instructions;

altering the arrangement of values in the registers;

performing a second plurality of weighted-rotation operations on the values in the registers;

again altering the arrangement of the values in the registers;

performing a third plurality of weighted-rotation operations on the values in the registers;

yet again altering the arrangement of the values in the registers; [[and]]

performing a fourth plurality of weighted-rotation operations on the values in the registers to obtain C intermediate floating point values; and

storing the C intermediate floating point values into a next available row of an intermediate buffer.

17. (Currently Amended) A carrier medium comprising software instructions executable by a microprocessor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions to implement a method of performing a two-dimensional discrete cosine transform (DCT), wherein the method comprises:

receiving a two-dimensional block of integer data having C columns and R rows, wherein each of the R rows contains a set of C row data values, wherein the block of integer data is indicative of a portion of an image, wherein each of C and R is an even integer; and

for each row,

loading the <u>entire set of</u> C row data values <u>of the row</u> into <u>a set of C/2</u> registers <u>of the microprocessor</u>;

converting the C row data values into floating point form, wherein <u>each of</u> the registers <u>holds</u> [[each hold]] two <u>of the</u> floating point row data values; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations are performed using SIMD floating point instructions;

altering the arrangement of values in the registers;

performing a second plurality of weighted-rotation operations on the values in the registers;

again altering the arrangement of the values in the registers;

performing a third plurality of weighted-rotation operations on the values in the registers;

yet again altering the arrangement of the values in the registers; and

performing a fourth plurality of weighted-rotation operations on the values in the registers to obtain <u>C</u> intermediate floating point values<u>; and</u>

<u>storing the C intermediate floating point values into a next available row of an intermediate buffer</u>.

18. (Currently Amended) A computer system comprising:

a processor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions; and

a memory coupled to the processor, wherein the memory stores software instructions executable by the processor to implement the method of receiving a <u>two-dimensional</u> block of integer data having C columns and R rows, wherein the <u>two-dimensional</u> block of integer data is indicative of a portion of an image; and

for two columns at a time,

loading column data <u>from the two columns</u> into registers of the processor <u>so that each of the registers holds one value from a first of the two</u>

columns and one value from a second of the two columns, wherein the one value from the first of the two columns and the one value from the second of the two columns are taken from the same row of the two-dimensional block;

converting the column data into floating point form [[, wherein the registers each hold a floating point column data value from two columns]]; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations for the two columns are performed in parallel using SIMD floating point instructions;

as each weighted-rotation operation is done, storing weighted-rotation operation results to an intermediate buffer.

19. (Currently Amended) A carrier medium comprising software instructions executable by a microprocessor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions to implement a method of performing a discrete cosine transform (DCT), wherein the method comprises:

receiving a two-dimensional block of integer data having C columns and R rows, wherein the two-dimensional block represents a portion of an image; and

for two columns at a time,

loading column data from the two columns into registers of the microprocessor so that each of the registers holds one value from a first of the two columns and one value from a second of the two columns, wherein the one value from the first of the two columns and the one value from the second of the two columns are taken from the same row of the two-dimensional block;

converting the column data into floating point form [[, wherein the registers each hold a floating point column data value from two columns]]; and

performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations for the two

columns are performed in parallel using SIMD floating point instructions;

as each weighted-rotation operation is done, storing weighted-rotation operation results to an intermediate buffer.

20. (Cancelled) ~~A method of performing a discrete cosine transform (DCT) using a microprocessor having an instruction set that includes single-instruction multiple-data (SIMD) floating point instructions, wherein the method comprises:~~

~~receiving a block of integer data having C columns and R rows; and~~

~~for two columns at a time,~~

~~loading column data into registers;~~

~~converting the column data into floating point form, wherein the registers each hold a floating point column data value from two columns; and~~

~~performing a plurality of weighted-rotation operations on the values in the registers, wherein the weighted-rotation operations for two columns are performed in parallel using SIMD floating point instructions~~.

21. (New) The method of claim 1, wherein C=8 and R=8.

22. (New) The method of claim 1, wherein each of the weighted rotations of said plurality, said second plurality, said third plurality and said fourth plurality have a computational form given by the expressions:

$$Y0 = A*X0 + B*X1,$$

$$Y1 = -B*X0 + A*X1,$$

wherein A and B are coefficients, X0 and X1 are inputs to the weighted rotation, Y0 and Y1 are results of the weighted rotation.